IN THE CLAIMS

1. (currently amended) An in-circuit emulation system, comprising:

a microcontroller;

a virtual microcontroller coupled to and executing instructions in lock-step

with the microcontroller, and wherein the microcontroller sends I/O read data to

the virtual microcontroller;

the virtual microcontroller having means for detecting a sequence of

instructions comprising an I/O read instruction followed by a conditional jump

instruction, and for computing a conditional jump address prior to receipt of I/O

read data from the microcontroller; and

the virtual microcontroller further having means for determining after

receipt of the I/O read data from the microcontroller whether to proceed with

instruction execution at a next consecutive address or at the conditional jump

address.

2. (original) The apparatus according to claim 1, wherein the conditional

jump address is computed while the I/O read data are sent from the

microcontroller to the virtual microcontroller.

3. (original) The apparatus according to claim 1, wherein the microcontroller

sets a zero flag if an I/O read test condition is met.

CYPR-CD01207 Serial No. 10/002,217 Page 2

Examiner: Proctor, J.

4. (original) The apparatus according to claim 3, wherein the jump condition

is met if the zero flag is set.

The apparatus according to claim 1, wherein the virtual 5. (original)

microcontroller is implemented in a Field Programmable Gate Array.

(currently amended) In an in-circuit emulation system having a 6.

microcontroller coupled to and operating in lock-step with a

microcontroller, a method of handling conditional jumps in the virtual

microcontroller, comprising:

detecting a sequence of instructions comprising an I/O read instruction

followed by a conditional jump instruction;

computing a conditional jump address prior to receipt of I/O read data

from the microcontroller; and

determining after receipt of the I/O read data from the microcontroller

whether to proceed with instruction execution at a next consecutive address or at

the conditional jump address.

7. (original) The method according to claim 6, further comprising executing

a next consecutive instruction in the event a conditional jump condition is not

met.

CYPR-CD01207 Serial No. 10/002,217 Page 3

Examiner: Proctor, J.

8. (original) The method according to claim 6, further comprising executing

an instruction at the conditional jump address in the event the conditional jump

condition is met.

9. (original) The method according to claim 6, wherein the conditional jump

address is computed while the I/O read data are sent from the microcontroller to

the virtual microcontroller.

10. (original) The method according to claim 6, wherein the microcontroller

sets a zero flag if an I/O read test condition is met.

11. (original) The method according to claim 10, wherein the jump condition

is met if the zero flag is set.

12. The method according to claim 6, wherein the virtual (original)

microcontroller is implemented in a Field Programmable Gate Array.

13. (original) The method according to claim 6, stored as instructions stored

in an electronic storage medium for execution as program steps on a

programmed processor forming a part of the virtual microcontroller.

14. (currently amended) In an in-circuit emulation system having a device

under test coupled to and operating in lock-step with a virtual processor, a

method of handling conditional jumps in the virtual processor, comprising:

CYPR-CD01207 Serial No. 10/002,217 Page 4

Examiner: Proctor, J.

detecting a sequence of instructions comprising an I/O read instruction

followed by a conditional jump instruction;

computing a conditional jump address prior to receipt of I/O read data

from the virtual processor; and

determining after receipt of the I/O read data from the device under test

whether to proceed with instruction execution at a next consecutive address or at

the conditional jump address.

15. (original) The method according to claim 14, further comprising executing

a next consecutive instruction in the event a conditional jump condition is not

met.

16. (original) The method according to claim 14, further comprising executing

an instruction at the conditional jump address in the event the conditional jump

condition is met.

17. (original) The method according to claim 14, wherein the conditional jump

address is computed while the I/O read data are sent from the device under test

to the virtual processor.

18. (original) The method according to claim 14, wherein the device under

test sets a zero flag if an I/O read test condition is met.

CYPR-CD01207 Serial No. 10/002,217 Page 5

Examiner: Proctor, J.

- 19. (original) The method according to claim 18, wherein the jump condition is met if the zero flag is set.
- 20. (original) The method according to claim 14, wherein the virtual processor is implemented in a Field Programmable Gate Array.

Examiner: Proctor, J.